Demilie and Salau
Journal of Cloud Computing (2022) 11:48
https://doi.org/10.1186/s13677-022-00299-1

Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH Open Access

Automated all in one misspelling detection and correction system for Ethiopian languages

Wubetu Barud Demilie[1*] and Ayodeji Olalekan Salau[2]

Abstract

In this paper, a misspelling detection and correction system was developed for Ethiopian languages (Amharic, Afan Oromo, Tigrinya, Hadiyyisa, Kambatissa, and Awngi). For some of these languages, there have been few works on typo detection and correction systems. However, an effective and all-in-one typo detector and corrector system for Ethiopian languages have yet to be developed. A dictionary-based methodology is used to detect and rectify various forms of misspelling-related issues. The major characteristics of the proposed model can be outlined by presenting suggestions for detected flaws and automatically correcting them utilizing the first suggestion. In addition, the proposed model is evaluated using dictionary-based data sets for all languages. The corpora used were gathered from a variety of sources, including economic, political, social, and related publications, newspapers, and magazines. In this model, the users can perform all spelling-related issues within a single system (all-in-one). That means if the user(s) is (are) working on the Amharic language and then he/she/they can change the language she/he/they prefer(s) without shifting to another graphical user interface (GUI). Here, the users can save time and perform their tasks easily. Similarly, the user(s) can improve their skills in the selected languages accordingly. Finally, precision, recall, and f-measures for each language have been computed following a successful evaluation of the model. The system outperforms an f-measure of 89.57%, 87.57%, 88.31%, 86.83%, 81.83%, and 87.59% for Amharic, Afan Oromo, Tigrinya, Hadiyyisa, Kambatissa, and Awngi languages respectively. Furthermore, recommendations have been provided for future researchers.

Keywords: Error correction, Error detection, Ethiopian languages, Dictionary, Spelling corrector, Suggestion

Introduction

It is fairly common to find numerous spelling errors in typed writings in various languages. The vast majority of people who type Amharic, Afan Oromo, Tigrinya, Hadiyyisa, Kambatissa, and Awngi make mistakes without realizing it. A devoted individual reads through written papers in various printing presses, discovers misspelled words, and corrects them. Spelling errors on printed sheets are still pretty common even after this method. This involves the development of an accurate and reliable misspelling detection and correction system that can be integrated with word processing tools.

Linguistic resources are vital in the development of natural language processing applications (NLPA). However, "low-resource" languages, primarily African languages, lack such tools and resources [1]. Despite this, Ethiopian languages remain among the world's "low-resource" languages, lacking the tools and resources required for NLPA and other techno-linguistic activities. However, a lack of appropriate datasets and good word embeddings have made it difficult to develop different systems that are reliable enough [1].

Different misspelling detection and correction systems have been developed by various professionals in the field for both foreign and Ethiopian languages. However, no one has developed such systems for more than five Ethiopian languages [2]. In this study, the researchers have used a dictionary-based model for spelling error detector and corrector to develop, build, and test an end-to-end

*Correspondence: wubetubarud@gmail.com; wubetuB@wcu.edu.et

[1] Department of Information Technology, Wachemo University, Hossana, Ethiopia
Full list of author information is available at the end of the article

system for six Ethiopian languages. Except [2], which introduced a misspelling detector and corrector system for five Ethiopian languages in a single system, no one has developed a misspelling detector and corrector system for more than three Ethiopian languages in a single system. As a result, the purpose of this research is to develop a misspelling detector and corrector system for six Ethiopian languages, which will serve as a first model for future researchers.

A feature in the proposed system alert users to select the proper language. Even if this work serves as a benchmark for Ethiopian languages, the paper provides the following contributions:

- ✓ We have presented a method for creating typographic errors for the six Ethiopian languages, which has never been done in a multilingual situation before (all earlier methods have either been for foreign languages or were language-specific except [2]).
- ✓ We have designed a system for six Ethiopian languages, within a single system. As a result, system users can switch between languages within a single system based on their interests.
- ✓ We have demonstrated the system's real-time usefulness in detecting and correcting errors by displaying its timely performance for each stage in the process.
- ✓ The system that has been developed outperforms existing dictionary-based and widely used typo detection and correction systems.
- ✓ The developed system demonstrates that the system is highly reliable.

This paper is organized into different but interrelated subtopics. The topic starts by discussing the basics of Ethiopian languages (i.e., the phonology, morphology, writing system, and word order) in Section 2, the related works in Section 3, the methodology in Section 4, results and discussions in Section 5, and conclusion and recommendation in Section 6.

The basics of Ethiopian languages

The languages of Ethiopia are classified into four major groups: Semitic, Cushitic, Omotic, and Nilo-Saharan [3–8]. The Semitic families are the most widespread languages in the country, with two of the most widely spoken languages being Amharic and Tigrigna [4].

The Ethio-Semitic languages are distinguished by the use of the Geez script, which is unique to Ethiopia and is known as Fidel (ፈደል). Ethiopians speak a variety of languages from the Cushitic language families like Afan Oromo, Hadiyyisa, Kambatissa, and Awngi [8] with the writing system as a Latin script for Afan Oromo, Hadiyyisa, and Kambatissa languages while Ethiopic alphabet (though there is some variation in the way it is written) is for Awngi language [9, 10]. Of all the language families, Afan Oromo is the most widely spoken language in Ethiopia [11, 12].

The phonology of Amharic, Tigrigna, and Awngi languages

Many phonetic sounds are shared by Amharic, Tigrigna, and Awngi languages. Tigrigna has 38 phonemes compared to Amharic's 34–35 (27–28 consonants and 7 vowels) (31 consonants and 7 vowels) [4, 6, 7, 13]. Although these languages share many phonetic features, Tigrigna has four sounds that Amharic does not have. Long consonants, also known as geminated consonants, are pronounced and contribute to semantic distinction in these languages. The Awngi language consists of 35 characters [10]. These characters are divided into two basic groups: vowels and consonants. The language has 6 vowels and 29 consonants.

The phonology of Afan Oromo, Hadiyyisa, and Kambatissa languages

They belong to the same language family and share phonetic characteristics such as the use of long and short vowels [3–5, 12, 14, 15]. Afan Oromo language has 24 to 28 consonant phonemes depending on the dialect and five vowels which all contrast with long and short vowels. Sometimes there is a change in vowel quality when the vowel is short. Short vowels tend to be more centralized than their counterparts [16, 17]. Though sometimes diphthongs may occur, none occur in a word's unaltered form. The Hadiyyisa language consists of 5 vowels and 23 consonants [14]. The consonant inventory encompasses 24 consonants and has 5 vowel systems for the Kambatissa language [15].

All these languages share many consonants and vowels despite each having its inventory of vowels and consonants accordingly. Of course, each has its own set of consonants. Almost all of the consonants in these languages are found in both single and geminated forms. The use of tones is another common phonetic feature of these languages, which makes all tonal languages. The detailed attempts to describe the phonology of the Awngi language have been presented in [10, 18].

The morphology

Morphologically, all the six languages can be considered complex [3, 6, 10, 12, 14, 15, 19]. Amharic and Tigrigna languages use the root and pattern system, which reflects their Semitic language morphology [22, 23]. In these languages, a root (also known as a radical)

is a set of consonants that bears the basic meaning of the lexical item, whereas a pattern is a set of vowels that are inserted between the root's consonants. These vowel patterns, when combined with affixes, yield derived words. Because of this derivational process, these languages are morphologically complex.

Aside from morphological information, some syntactic information is also expressed at the word level. Furthermore, an orthographic word may include syntactic words such as prepositions, conjunctions, negation, and so on, resulting in a variety of word forms. Nouns are inflected for number, gender, definiteness, and cases in these languages, whereas verbs are inflected for person, number, gender, tense, aspect, and mood. Nominals in Afan Oromo, Hadiyyisa, Kambatissa, and Awngi Languages like those in Semitic languages, are inflected for number, gender, case, and definiteness, while verbs are inflected for person, number, gender, tense, aspect, and mood [3, 9, 10, 15, 18, 20, 21].

The writing system

The Ethiopic script known as Fidel is used to write Amharic, Tigrigna, and Awngi [4–7, 10, 15]. This script is syllabic because each symbol represents a consonant combined with a vowel, and the vowel does not exist on its own. In other words, each orthography symbol represents a consonant-vowel (CV) syllable. Each of the core consonants has 7 shapes or orders for Amharic and Tigrigna, and 6 for Awngi [9, 10] based on the vowels that are combined with them. Even though long consonants are pronounced in these languages and the writing system does not distinguish between short and long consonants.

The Latin script is used in the Afan Oromo, Hadiyyisa, and Kambatissa writing systems. The current writers distinguish between geminated and non-geminated consonants in all languages. Long and short vowels are also indicated in the writing system.

Word ordering

Ethiopian languages are morphologically rich to the point where words must be further segmented or a single word from these languages is aligned with as many as a handful of words in languages such as English [7, 10, 19].

The word order in the languages under consideration differs. In this regard, Amharic, Afan Oromo, Tigrigna, Hadiyyisa, Kambatissa, and Awngi have subject object verb (SOV) structure [6, 10, 14, 15, 21–23]. The various word orders used in these languages pose a significant challenge for multilingual machine translation systems. Another difficulty is the existence of word order flexibility. For example, even though the Afan Oromo language uses the SOV word order format, nouns can be changed based on their role in a sentence, making the word order flexible.

Related works

There have been numerous studies on the topic of spelling error detection and correction in the absence of widely accepted standards. Each research work establishes a benchmark and compares it to a (typically limited) selection of methods and/or algorithms. A model was proposed and characterized in research work [2] as offering suggestions for detected flaws and automatically correcting them using the first suggestion. For the languages Amharic, Afan Oromo, Tigrinya, Hadiyyisa, and Awngi, the researcher had presented his work with precision (86.6%, 85.3%, 83.9%, 82.8%, and 84.7%), recall (84.7%, 81.9%, 82.4%, 81.6%, and 81.9%), and f-measure (85.65%, 83.6%, 83.15%, 82.2%, and 83.3%) respectively.

The work [24] shows how to use an automatic spelling corrector for the Amharic language. They have used a corpus-driven technique with the noisy channel for spelling correction. To infer linguistic expertise, a text corpus was used. The proposed method was easily adapted to other written languages, as long as it was typed using a QWERTY keyboard with direct keystroke-to-character mappings. Because Amharic language letters are syllabic, they employ a modified version of the scheme for Ethiopic Representation in the American Standard Code for Information Interchange (ASCII) for transliteration, as do most Amharic language keyboard input techniques. The proposed approach was assessed using Amharic and English language test data, and it outperformed the baseline systems, GNU Aspell and Hunspell. The smoothed language model, the generalized error model, and the ability to consider the context of misspellings all contributed to a superior result. Furthermore, they have used a phrase list constructed from commonly occurring terms in a text corpus to detect spelling errors rather than a handcrafted vocabulary. Aside from being simple to compile, a term list like this has the added benefit of being able to handle uncommon terms, proper nouns, and neologisms. Finally, they have suggested that they tried to analyze their approach for real-world spelling errors in future work.

The work in [25] designed and developed a non-word Afan Oromo language spell checker system. The system was designed using morphological analysis and a dictionary look-up (i.e. morphology-based spell checker). The knowledge of language morphology was required to construct the morphology-based spell checker. As a result, the Afan Oromo language's morphological features were investigated. The effort was the first of its sort for

the Afan Oromo language, to the best of the researcher's knowledge. The research work of [26] uses a rule-based system from a linguistic and computational standpoint, Afan Oromo language analyzer for spell checker determined that building language application is a resource-intensive activity that demands the active participation of stakeholders.

According to the researcher's investigation, the algorithm and approaches utilized in the study were performed well. The findings inspire future research in the field, particularly intending to produce a full-fledged Afan Oromo language spell checker. The work in [27] presents the development of a spelling corrector for the Amharic language. The technique was created with tolerant-retrieval search systems in mind. To provide appropriate solutions for misspelled words, the spelling corrector employs Amharic language Megaphone and edit distance algorithms. For that purpose, a standard dictionary was created for the spelling corrector to use.

Application of the spelling corrector to a test data corpus indicates that the algorithm makes 81.7% valid suggestions. A useful spell checker system was developed for the Amharic language in [28]. As a result, using the Visual Basic (VB) programming language, a functional spell-checking model was built and implemented. The Megaphone algorithm was used to detect spelling errors, and the edit distance algorithm was used to select the most likely correct word for the misspelled terms. Finally, the prototype was placed to the test using 500 words of test data, 100 of which were purposefully misspelled, and a lexicon containing 125,000 terms. The system performs admirably in terms of error detection and suggestions during the testing process.

According to [29], researchers claim that using the n-gram model can improve the performance of the spelling corrector system, and it can be used for a range of languages. The "n-grams can be used in two ways: without a lexicon or with a vocabulary," was suggested to those researchers. According to the researchers' investigation, the spell corrector's performance was limited without a dictionary. Its main advantage is that it is straightforward and does not necessitate the use of a dictionary. If two words are close together, it can be used to define the distance between them, and the words are always checked against the dictionary. Finally, they concluded that combining the two models will increase the spelling corrector system's performance.

Authors in [30] investigated the feasibility of constructing and building an Amharic language morphology-based spelling error detection and correction system. According to the investigation, 717 morphological rules were created for testing, 2398 stem words from each root word category were stored, and the system was evaluated using 1724 words from various derivational and inflectional categories. A prototype was created by utilizing the Python programming language and three CSV-formatted knowledge bases. Precision, recall, and predictive accuracy was utilized to evaluate the system. For accurately recognized invalid words, the experimental findings revealed 96% lexical recall, 89% error recall, 99% lexical precision, 70% error precision, 95% predictive accuracy, and 70% corrections. According to the researchers, the result indicates that the algorithm has a high level of accuracy in classifying terms as valid or invalid, but that suggestion generation should be improved. In [31], the n-gram technique was utilized to construct a language-independent spelling corrector. It was used to find and fix spelling mistakes.

According to the researcher, the n-gram model corrects and suggests by selecting the most appropriate suggestions from a list of corrective suggestions based on lexical resources and n-gram data. Finally, the researcher was able to gather and infer a total performance of 93%. The work of [32] designed a prototyping process model for the building of the Tigrigna language spelling checker and corrector in Android cellphones. In mobile phone devices such as smartphones, a spell checker and corrector system for the low-resource Tigrigna language was developed. The corpus used in this study contained 430,379 Tigrigna language words. To show the correctness of the planned spellchecker and corrector model and algorithm, a prototype was developed.

The prototype for Tigrigna's language spell checker and correction system for mobile phones was tested, and the prototype was judged to be 92% accurate. When writing Tigrigna language words on mobile phones, this experiment shows that the system model is effective at spell checking and suggesting relevant proper terms, as well as reducing misspelled input words.

A new strategy for detecting and correcting space deletion problems has been proposed by researchers in [33]. An altered version of the Levenshtein distance was employed to correct the flaws. To analyze the chosen technique, they have used a language produced from artificial intelligence Matane media reports. The results of the study demonstrated that the technique used was beneficial. The work of [34] has developed a way for correcting word spelling problems in English language texts. A combined spell correction method was proposed, which included Levenshtein distance for comparing misspelled words to correctly spelled words in dictionaries, an improved Double Megaphone algorithm for Chinese English learners with vowel phoneme rule sets, and

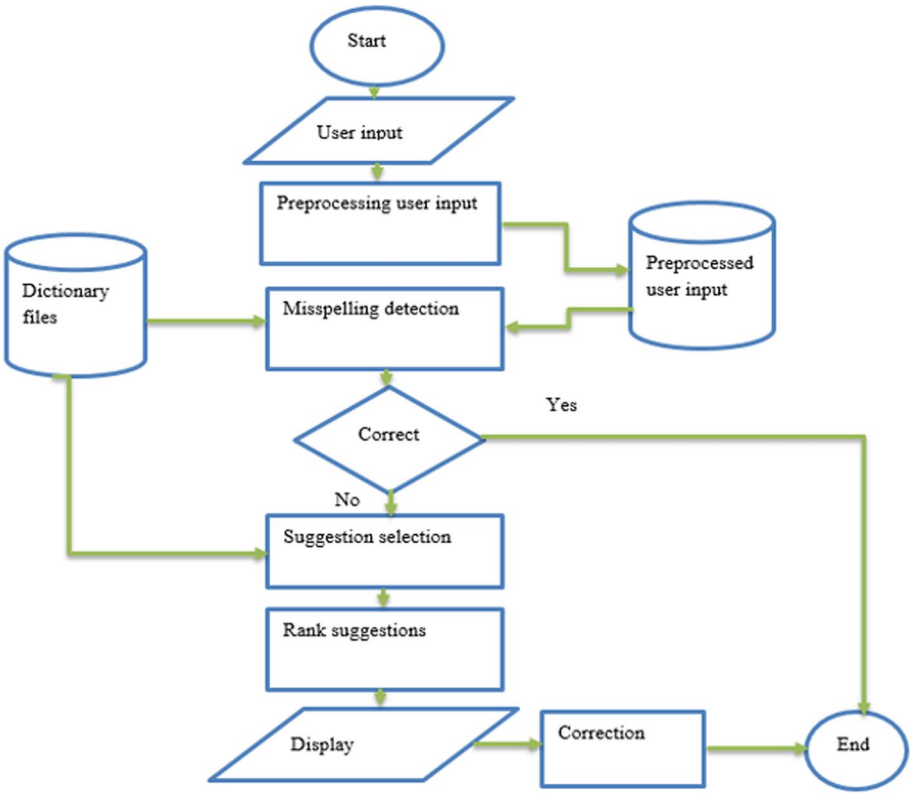

Fig. 1 Workflow of the proposed system

global vectors (GloVe) for character representation that can generate vectors to obtain letter misspelled word suggestion lists.

The combined approach was proposed and found to be superior to phonetic correction or the edit distance method alone, and an experiment was conducted to compare this approach to two commonly used spelling checker tools, which revealed that the approach was superior to them in correcting misspelled words, and the success rates of suggestion lists for spelling mistakes were on target.

The Damerau-Levenshtein algorithm and the n-gram were used by the authors in [35] to create a spelling error correcting system for the Amazigh language. The system recommended probable words for each misspelled word in the text. Successful testing was carried out using an Amazigh language corpus. The system's performance was assessed independently for misspelling detection and correction. Similar to the other approaches, their system produced f-measures ranging from 86.62% to 98.74% when it came to typo detection. While the precision of the adjustment has shown to be satisfactory when compared to other approaches.

Methodology

There are a variety of methods for detecting and correcting spelling mistakes in written documents. The researchers employed a dictionary-based strategy to associate and detect input strings in a dictionary, lexicon, corpus, or a combination of lexicons and corpora for the study.

The datasets or lexicon files for the six Ethiopian languages were compiled with the help of linguistic experts from various genres that contain balanced corpora and/or lexicon. After collecting balanced corpora from different genres and text preprocessing mechanisms have been applied. Here, text preprocessing, spell checking, and spelling suggestions are the three key aspects of the proposed system as indicated in Fig. 1.

The system accepts user input, preprocesses it to extract the root word and required features, then checks the word's validity against the dictionary file. If the word is found to be incorrect, the system will provide a list of possible correct words that are closer to the user's input by calculating the edit distance between the user input word and the dictionary word. The model shown in Fig. 1 depicts the general step that the system will follow to validate the accuracy of the user

Table 1 Word dictionaries for the languages

No.	Language	Amount of words (dictionary files)
1.	Amharic	1,009,072
2.	Afan Oromo	882,328
3.	Tigrinya	1,003,176
4.	Hadiyyisa	982,328
5.	Kambatissa	879,328
6.	Awngi	693,121

input and to provide the closest possible list of terms as a correction for the misspelled words.

User inputs may contain numbers, punctuation, or words from these languages, as shown in Fig. 1. As a result, before moving on to the misspelling detection phase, the algorithm filters out such inputs. The output of the phase is used to determine the word validity. The three main responsibilities in this phase are punctuation removal, normalization, and stem which are used to automatically generate keys.

According to the model, the system will remove punctuations from the user's input using a punctuation removal algorithm. The input word is compared to a list containing the list of possible punctuation marks utilized in the method. Furthermore, misspellings in all Ethiopian languages are caused by the availability of several alphabets with similar sounds and shapes according to the structure of the languages. As a result, it is planned that those letters will be combined into a single common representation to detect and repair errors caused by linguistic features.

The proposed system will look for these letters in user input and replace them with their standard representation. Stemming has been incorporated with the misspelling detection and correction methods, which finds the stemming of a word as a key for detecting similarity, in addition to the preprocessing techniques mentioned above. The researchers have used balanced test corpora which were collected from different sources for the study as indicated in Table 1.

Results and discussions

The evaluation results for Amharic, Afan Oromo, Tigrinya, Hadiyyisa, Kambatissa, and Awngi languages are reported in terms of misspelling detection and correction capabilities. To evaluate the system's performance, the researcher collected corpora that were greater than [2], which are from balanced sources, and within detailed linguistic experts of the languages.

All languages were evaluated using the corpora after the study's suitable and extensive preprocessing methods, as shown in Fig. 1. To begin, we have used the Amharic language test data from the dictionary files to conduct an examination. Second, we looked at the results of the Afan Oromo language test, which is included in the dictionary file list. Thirdly, the dictionary files were used to conduct an evaluation utilizing Tigrinya language test data. Fourth, we have used the Hadiyyisa language test data from the dictionary file list to perform an evaluation. Fifthly, we have used Kambatissa language test data from the dictionary file list to conduct an evaluation. Finally, we have used Awngi language test data from the dictionary file list to conduct an evaluation.

The relative locations of the correct spellings in the reasonable recommendations list were used to evaluate the proposed system. The criteria for evaluation are the capacity to detect misspellings and the quality of plausible alternatives provided for each misspelling of the selected Ethiopian languages. Similar to the binary classification of phrases as misspelling and appropriate term classes, precision, recall, and the f-measure is used to evaluate the system's capacity given by Eqs. (1–3).

As a result, any misspelling detector and corrector system would need to have a precision of 100% to identify all misspellings and only misspellings, as well as a recall of 100% to recognize all valid words as correct and all invalid words as misspellings [36]. As a result, recall is largely used to determine language coverage. The f-measure presents a high-level summary of the capabilities of the misspelling detector.

Manually created test data was used as the gold standard or balanced dataset for the evaluation to evaluate and compute the actual scores.

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{1}$$

$$Recall = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{2}$$

$$F - measure = \frac{2 * (\mathrm{Precision} * \mathrm{Recall})}{\mathrm{Precision} + \mathrm{Recall}} \tag{3}$$

The effective locations of accurate spellings in the dictionary proposals list, which has been developed to reward the excellence of suggestions received by the spelling corrector. The correct spelling correction should always be at the top of the priority list in the ideal situation. For the system, different experiments were carried out individually for each of the selected languages to evaluate the spelling corrector and the suggested system's quality. To accomplish this, the study was carried out in

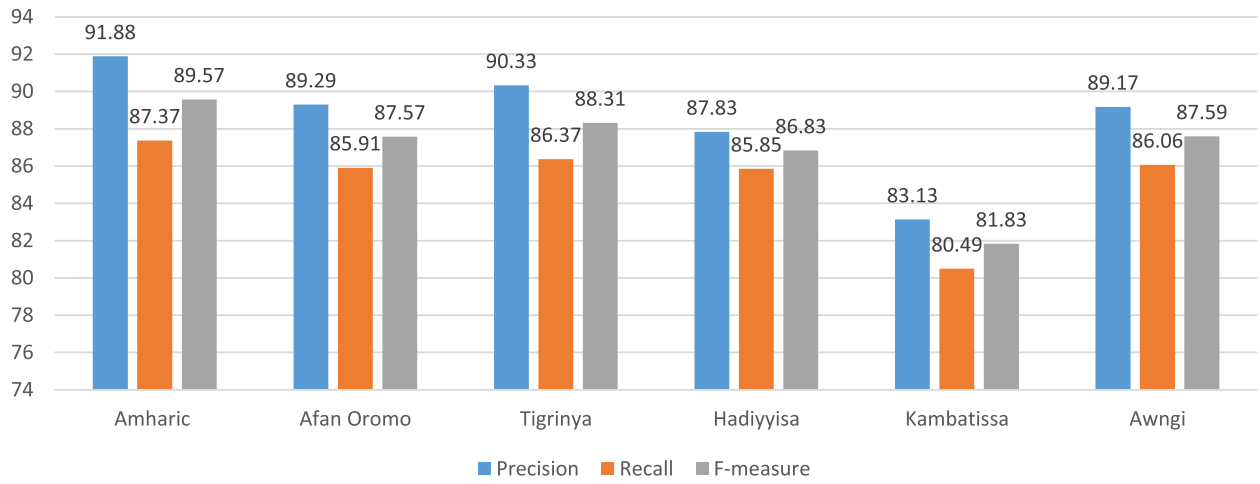

Fig. 2 Precision, recall, and f-measure results of the model

all the languages listed in Table 1. The Ge′ez script is used in languages such as Amharic, Tigrinya, and Awngi, and the researchers have incorporated a font identification in the system. However, the remaining three languages utilize Latin script, which can be used without any font changes.

The evaluation results of the proposed system are shown in Fig. 2. The evaluation of the results is presented in Table 2 and Fig. 2 indicates that the f-measure improves for the proposed system based on the collected corpora for the selected Ethiopian languages.

Table 3 shows the fragment java code which helps in the selection of the languages that the users are interested in. The following common java source code was considered for the model [1].

Accordingly, Fig. 3 shows the sample GUI for the proposed system. After this, if the user has selected "Amharic" language with Amharic texts (for example, "የመንግሥት ተቋም መዘጋት በተገቢው ሚዲያ ይፋ መደረግ አለበት።"), the written "Amharic" texts will have as of Fig. 4.

The words in Fig. 4 are underlined in **a red zigzag line**. Which indicates that all are not in dictionary file lists. So, the user should have to click on the underlined word (can use **"F7"** from the computer keyboard) to display the list of alternatives.

After this, the system user will see Fig. 5, in which terms not found in the dictionary file list are highlighted in red. Here, the user can choose from a variety of operations.

If the user adds all words to the dictionary (click on "**ወደ መዝገበ ቃላቱ ይጨምሩ**") or if the words are already in the dictionary files list, the words will not be emphasized with **a red zigzag line**, as seen in Fig. 6.

The researchers have tested the proposed system by creating commonly known errors in the languages. Since the collected and used dictionary files of each language were from different genres with the help of linguistic experts of each language, the system checks the errors easily and suggests the best alternative from the given list of words that have been provided in the dictionary. For example, if the user wants to replace the missing character from the word "**ትምህርን**" and if the word is not in the dictionary file list, Fig. 7 includes some of the suggestion lists.

The algorithm quickly checks the errors and provides the best replacement from the given list of terms that have been provided in the dictionary because the collected and used dictionary files of each language were from diverse genres with the help of linguistic experts of each language. Figure 7 illustrates some of the suggested

Table 2 Precision, recall, and f-measure results of the model

Metric	Languages and Evaluation Results					
	Amharic	Afan Oromo	Tigrinya	Hadiyyisa	Kambatissa	Awngi
Precision	91.88%	89.29%	90.33%	87.83%	83.13%	89.17%
Recall	87.37%	85.91%	86.37%	85.85%	80.49%	86.06%
F-measure	89.57%	87.57%	88.31%	86.83%	81.83%	87.59%

Table 3 Sample source code for the graphical user interface (GUI)

```
“import javax. Swing. JEditorPane;
import javax.swing. JFrame;
import javax.swing. JTextPane;
import com.inet.jortho. FileUserDictionary;
import com.inet.jortho. SpellChecker;
import java.awt. Font;
public class SampleApplication extends JFrame{
public static void main (String[] args){
new SampleApplication().setVisible(true);
}
private SampleApplication(){
super(“All in One Spelling Corrector/የስድስት ቋንቋዎችን ፊደል በአንድ መተግበሪያ ማስተካከያ
ስርዓት “);
JEditorPane text = new JTextPane();
Font font = new Font(““, Font.BOLD, 44);
text.setText(“All in One Spelling Corrector /የስድስት ቋንቋዎችን ፊደል በአንድ መተግበሪያ
ማስተካከያ ስርዓት “);
add(text);
text.setFont (font);
setSize(250, 180);
setDefaultCloseOperation(EXIT_ON_CLOSE);
setLocationRelativeTo(null);
SpellChecker.setUserDictionaryProvider (new FileUserDictionary());
SpellChecker.registerDictionaries(null, null);
SpellChecker.register(text);
}
}”
```

lists if the user wants to replace the missing character from the word “**ትምህርን**” and the word is not in the dictionary file list.

Here, if the system user wants to edit the dictionary files, he/she can click on “**መዝገበ ቃላቱን ያስተካክሉ**” then he/she will get Fig. 8 and he/she

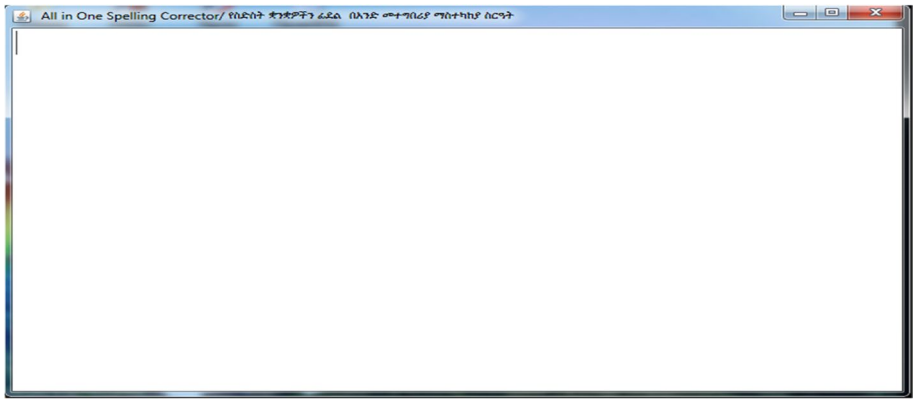

Fig. 3 GUI for the proposed system

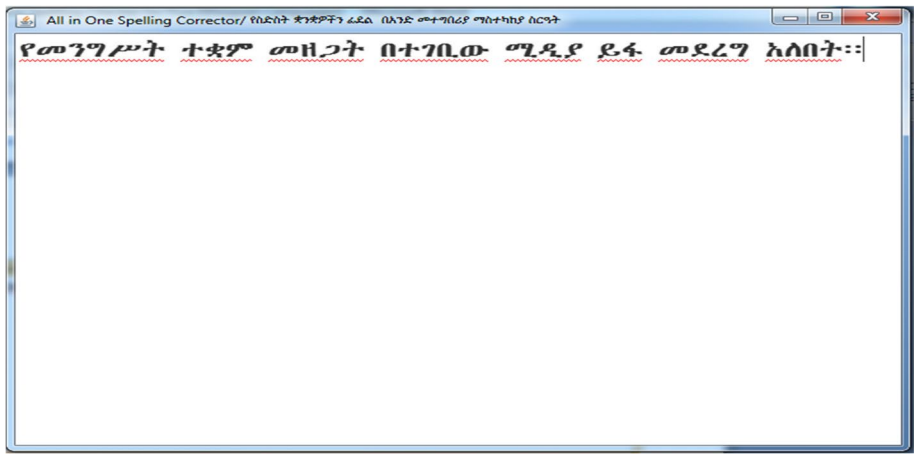

Fig. 4 List of words that are not in Amharic dictionary files

can do the remaining operations accordingly. Here, if the user selects "ትምህርት" according to the meanings of the sentence "በኮሮና በሽታ ምክንያት ተቋርጦ የነበረው ትምህርት መቾ እንደሚጀመር የታወቀ ነገረ የለም።።". The corrected sentence looks like "በኮሮና በሽታ ምክንያት ተቋርጦ የነበረው ትምህርት መቾ እንደሚጀመር የታወቀ ነገረ የለም።።".

Accordingly, as we know the words "መቾ (but it depends on the system user)" and "ነገረ" is not spelled correctly, but looks like a word that has been spelled properly as of Fig. 9. Such errors may result from improper word insertion into the dictionary (perhaps owing to a system user or developer error).

Finally, Fig. 10 shows terms that are devoid of any spelling errors, except for the erroneously added words "መቾ" and "ነገረ".

If the system user chooses "Afan Oromo" as the language, sample text will appear, such as "**Mana kadhata bira jiraatti**." Which translates to "She lives close to the church," as illustrated in Fig. 11.

Figure 11 shows that the word "**mana**" is in the dictionary file, but the words immediately adjacent to it are not in the list. By pointing to the exact word that has been underlined with a red zigzag line as shown in Fig. 12, the system user can see a list of suggestions. For example, there are some possibilities for the word "**jiraatti**".

Finally, Fig. 13 depicts the correctly spelled sentence "**mana kadhata bira jiraatti**." The users of the system can use the same procedure to detect and correct all spelling-related difficulties for all of the languages that have been selected.

Fig. 5 Sample GUI for words that are not in the dictionary

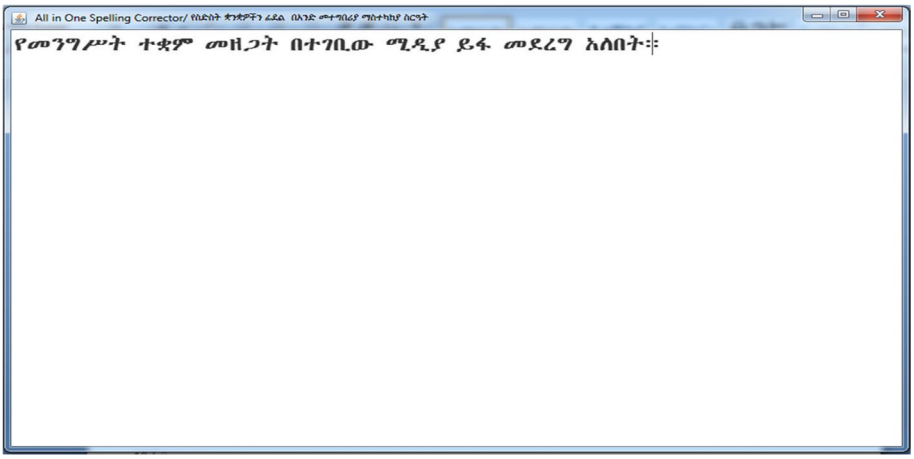

Fig. 6 Sample GUI for words that are free from any spelling error

Fig. 7 Sample GUI for word suggestion

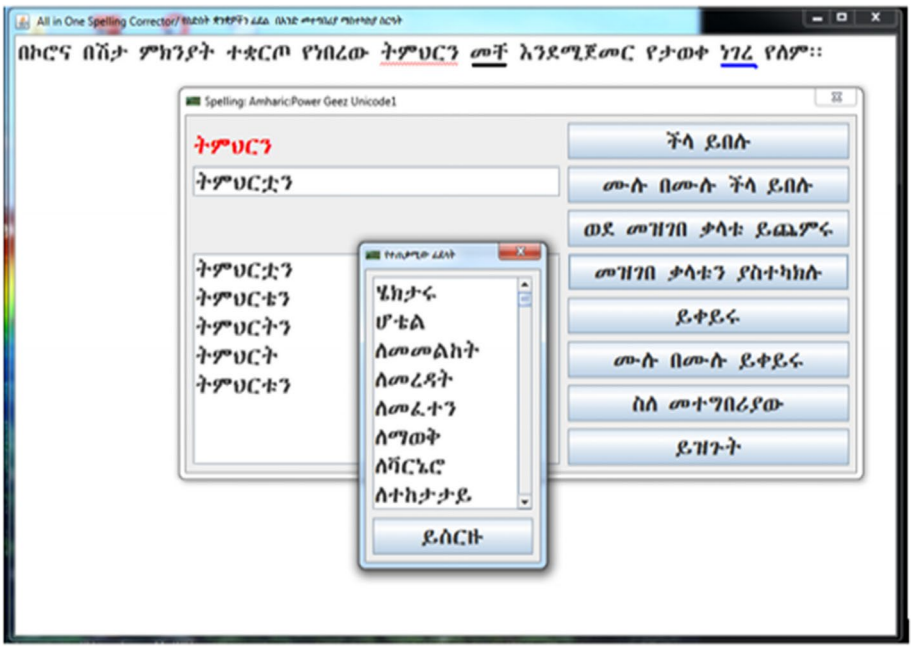

Fig. 8 Sample GUI model to edit the dictionary

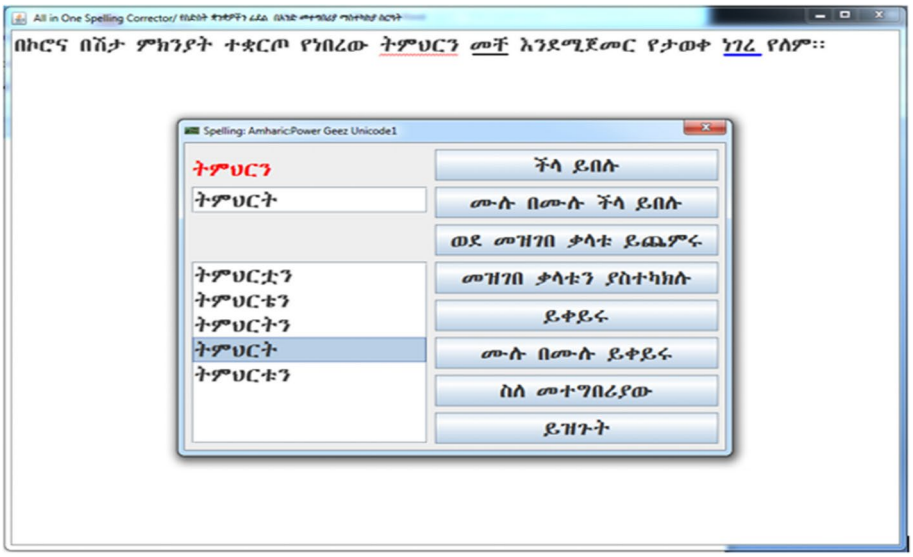

Fig. 9 Sample GUI model with a list of suggestions for a single word

Conclusion and recommendation

The terms in the lexical dictionary are heavily relied upon by misspelling detection and correction systems. It is a tool that must be built for all types of system users, and it is one of the natural language processing applications that detects and corrects problems in various applications. During spelling certain words have only a few words spelled similarly, even multiple errors will retrieve the correct term. Other terms will have a large number of words that are spelled in the same manner, making change difficult or impossible. The proposed model was created using a dictionary-based technique and it detects and corrects a wide range of spelling mistakes.

The key characteristics of the intended model can be outlined in terms of making suggestions for problems that have been detected and automatically correcting

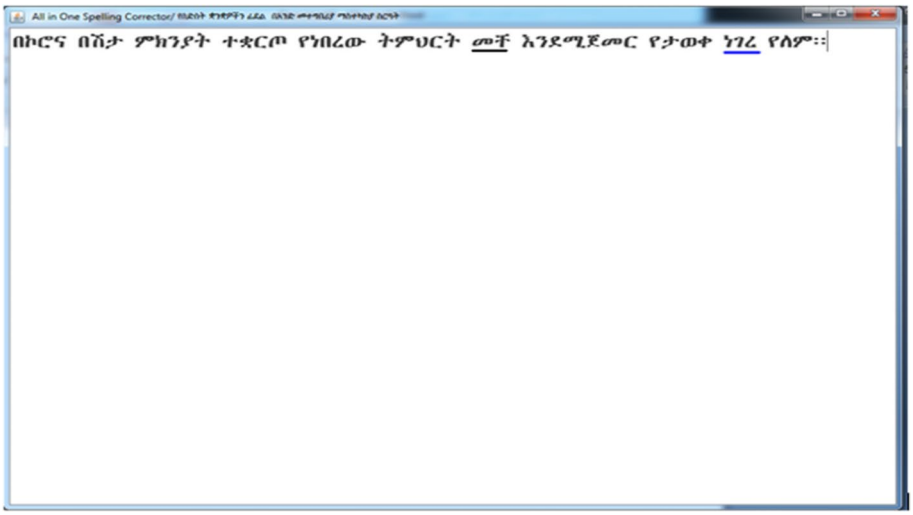

Fig. 10 Sample GUI for correctly spelled words

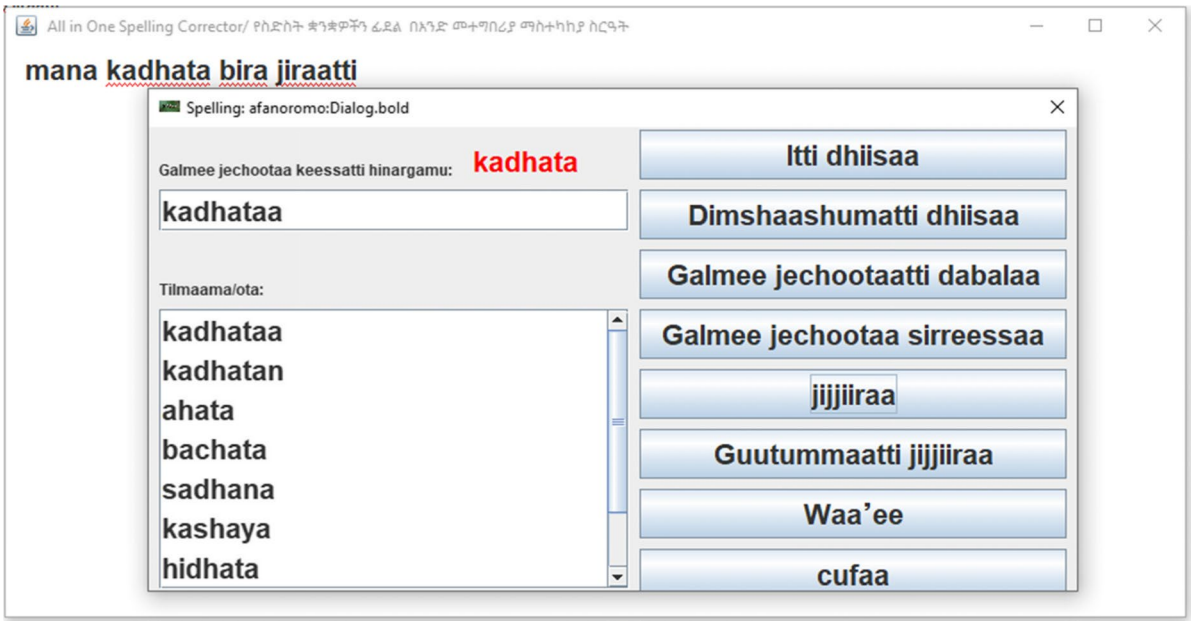

Fig. 11 Sample GUI for word suggestions from the Afan Oromo dictionary file

them utilizing the first suggestion. Due to the number of corpora collected, how preprocessing techniques were implemented, the places where the corpora were collected, and the number of linguistic experts that contributed, the suggested system outperforms better than the works. According to the results of the proposed model, the number of corpora gathered, the preprocessing techniques employed, the locations where the corpora were collected, and the number of linguistic experts participating all influence the performance of the systems. Finally, the proposed system performs an f-measure of 89.57%, 87.57%, 88.31%, 86.83%, 81.83%, and 87.59% for Amharic, Afan Oromo, Tigrinya, Hadiyyisa, Kambatissa, and Awngi languages respectively. Since this work is a benchmark for Ethiopian languages, the researcher recommends future researchers use other approaches to improve the performance of the system.

Fig. 12 Sample suggestion for the word "jiraatti"

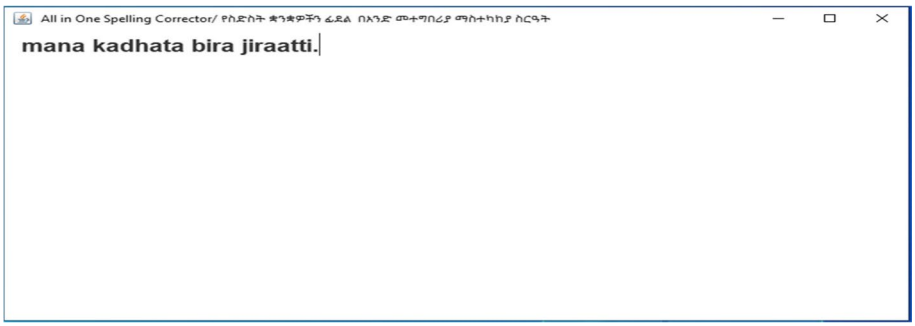

Fig. 13 Sample GUI for correctly spelled words

Acknowledgments
Not applicable.

Authors' contributions
Wubetu Barud Demilie: Prepared the manuscript including analysis, data curation, visualization, conceptualization, methodology, and writing of the original draft. **Ayodeji Olalekan Salau**: Performs the tasks including conceptualization, validation, writing, review, and editing of the final work. Both authors read and approved the final manuscript.

Funding
Not applicable.

Availability of data and materials
Not applicable.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that no competing interest.

Author details
[1]Department of Information Technology, Wachemo University, Hossana, Ethiopia. [2]Department of Electrical/Electronics and Computer Engineering, Afe Babalola University, Ado-Ekiti, Nigeria.

Received: 1 March 2022 Accepted: 24 July 2022
Published online: 24 September 2022

References

1. Gereme F, Zhu W, Ayall T, Alemu D (2021) Combating fake news in 'low-resource' languages: Amharic fake news detection accompanied by resource crafting. Inf 12(1):1–9. https://doi.org/10.3390/info12010020
2. Demilie WB (2020) Multilingual spelling checker for selected Ethiopian languages. Int J Adv Sci Technol 29(7):2641–2648
3. Endale Daba MM (2021) Improving Afaan Oromo question answering system: definition, list and description question types for non-factoid questions. pp 1–101
4. Abate ST, Tachbelie MY, Schultz T (2020) Deep Neural Networks Based Automatic Speech Recognition for Four Ethiopian Languages," ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and

Signal Processing (ICASSP), 2020, pp 8274-8278. https://doi.org/10.1109/ICASSP40776.2020.9053883.
5. Solomon Teferra Abate TS, Tachbelie MY (2021) End-to-end multilingual automatic speech recognition for less-resourced languages: the case of four ethiopian languages. CSL, University of Bremen, Germany. ICASSP 2021–2021 IEEE Int Conf Acoust Speech Signal Proc pp 7013–7017
6. Abate ST et al (2020) Large vocabulary read speech corpora for four Ethiopian languages: Amharic, Tigrigna, Oromo, and Wolaytta. Lr 2020 - 12th Int Conf Lang Resour Eval Conf Proc pp 4167–4171. https://doi.org/10.18653/v1/2020.winlp-1.5
7. Abate ST, Tachbelie MY, Schultz T (2020) Multilingual acoustic and language modeling for the-Semitic languages. Proc Annu Conf Int Speech Commun Assoc INTERSPEECH pp 1047–1051. https://doi.org/10.21437/Interspeech.2020-2856
8. LibGuide. Languages of Ethiopia - All About Ethiopia. 2021. https://liupalmer.libguides.com/c.php?g=1143777&p=8348034
9. Wikipedia. Awngi language - Wikipedia. https://en.wikipedia.org/wiki/Awngi_language
10. Misikir S, Tsegaye T (2013) Developing a stemming algorithm for Awngi text: the longest match approach. http://213.55.95.56/handle/123456789/14814?show=full
11. Gurmessa DK, Mamo G, Biru JD (2020) Afaan Oromo Text Content-Based Fake News Detection using Multinomial Naive Bayes. 01(01):26–36. https://www.researchgate.net/publication/347508623_Afaan_Oromo_Text_Content-Based_Fake_News_Detection_using_Multinomial_Naive_Bayes
12. Walga TK (2021) Prospects and challenges of afan Oromo: a commentary. Theory Pract Lang Stud 11(6):606–612. https://doi.org/10.17507/tpls.1106.03
13. Worku MH, Woldeyohannis MM (2022) Amharic Fake News Detection on Social Media Using Feature Fusion. Lect Notes Inst Comput Sci Soc Telecommun Eng LNICST 411 LNICST(January):468–479. https://doi.org/10.1007/978-3-030-93709-6_31
14. Sewasew. Sewasew _ Hadiyya language. https://en.sewasew.com/p/hadiyya-language-(%E1%8B%A8%E1%88%83%E1%8B%B5%E1%8B%AB-%E1%89%8B%E1%8A%95%E1%89%8B)
15. Jonathan Samuel Sumamo ST (2018) Designing a stemming algorithm for Kambaata text: a rule based approach. pp 1–119. http://197.156.93.91/bitstream/123456789/4455/1/Designing%20a%20Stemming%20Algorithm%20for%20Kambaata%20Text%20-%20A%20Rule%20Based%20Approach_Print%20Version3.pdf
16. Wikipedia. Oromo phonology - Wikipedia. https://en.wikipedia.org/wiki/Oromo_phonology
17. Tesema W, Tamirat D (2017) Investigating Afan Oromo language structure and developing effective file editing tool as plug-in into Ms. Word to support text entry and input methods. Am J Comput Sci Eng Surv pp 001–008
18. Joswig A (2010) The phonology of Awngi. SIL Electron Work Pap 2010–003:37
19. Abate ST et al (2018) Parallel corpora for bi-directional statistical machine translation for seven Ethiopian language pairs. Proc 27th Int Conf Comput Linguist pp 3102–3111
20. Wedekind C, Wedekind K (2002) Sociolinguistic survey of the Awngi language of Ethiopia. SIL Electron Surv Reports 2002–044:24
21. the free encyclopedia Wikipedia. Kambaata language - Wikipedia. https://en.wikipedia.org/wiki/Kambaata_language#:~:text=Kambaata%20is%20a%20Highland%20East,large%20amount%20of%20morphophonemic%20changes
22. Abb. Girma Manniso Waaxumo. Hadiyya (Hadiyyisa) Language Orthography - Alphabet and Writing - Themes on the Hadiya People of Ethiopia. https://hadiyajourney.com/hadiyya-hadiyyisa-language-orthography-alphabet-and-writing/
23. Mihret M, Atinaf M (2019) Sentiment analysis model for opinionated Awngi text. IEEE AFRICON Conf pp 1–7. https://doi.org/10.1109/AFRICON46755.2019.9134016
24. Gezmu AM, Nürnberger A, Seyoum BE (2018) Portable Spelling Corrector for a Less-Resourced Language : Amharic. pp 4127–4132
25. Ganfure GO, Midekso D (2014) Design And Implementation Of Morphology Based Spell Checker. 3(12):118–125
26. Jeldu MD, Mehta R (2018) Rule-based afan Oromo analyzer for spell checker 1 1,2. (7):36–39
27. Mekonnen A (2012) Development of an Amharic spelling corrector for tolerant-retrieval. Proc Int Conf Manag Emergent Digit Ecosyst MEDES 2012:22–26. https://doi.org/10.1145/2457276.2457281
28. Assefa G (2018) Automatic Amharic Spelling Error Detection and Correction Using. 5(6):605–611
29. Kumar R, Bala M, Sourabh K (2018) A study of spell checking techniques for Indian Languages. (March):105–113.
30. Shimelis M (2020) Amharic Spelling Error Detection and Correction System. pp 2–130
31. El Atawy SM (2018) Automatic Spelling Correction based on n-Gram Model. 182(11):5–9
32. Aray PU The construction of Tigrigna Spelling Checker and Corrector in android smartphones through prototyping process model. Afr J Online (Ajol) 15(02):1–15
33. Abdellah Y, Lhoussain AS, Hicham G, Mohamed N (2020) Spelling correction for the Arabic language-space deletion errors. Procedia Comput Sci 177:568–574. https://doi.org/10.1016/j.procs.2020.10.080
34. Huang G, Chen J, Sun Z (2020) A correction method of word spelling mistake for English text. J Phys Conf Ser 1693(1):2–18. https://doi.org/10.1088/1742-6596/1693/1/012118
35. Chaabi Y, Ataa Allah F (2021) Amazigh spell checker using Damerau-Levenshtein algorithm and N-gram. J King Saud Univ - Comput Inf Sci. https://doi.org/10.1016/j.jksuci.2021.07.015
36. Gezmu AM, Nürnberger A, Seyoum BE (2019) Portable spelling corrector for a less-resourced language: Amharic. Lr 2018 - 11th Int Conf Lang Resour Eval (May):4127–4132

Publisher's Note